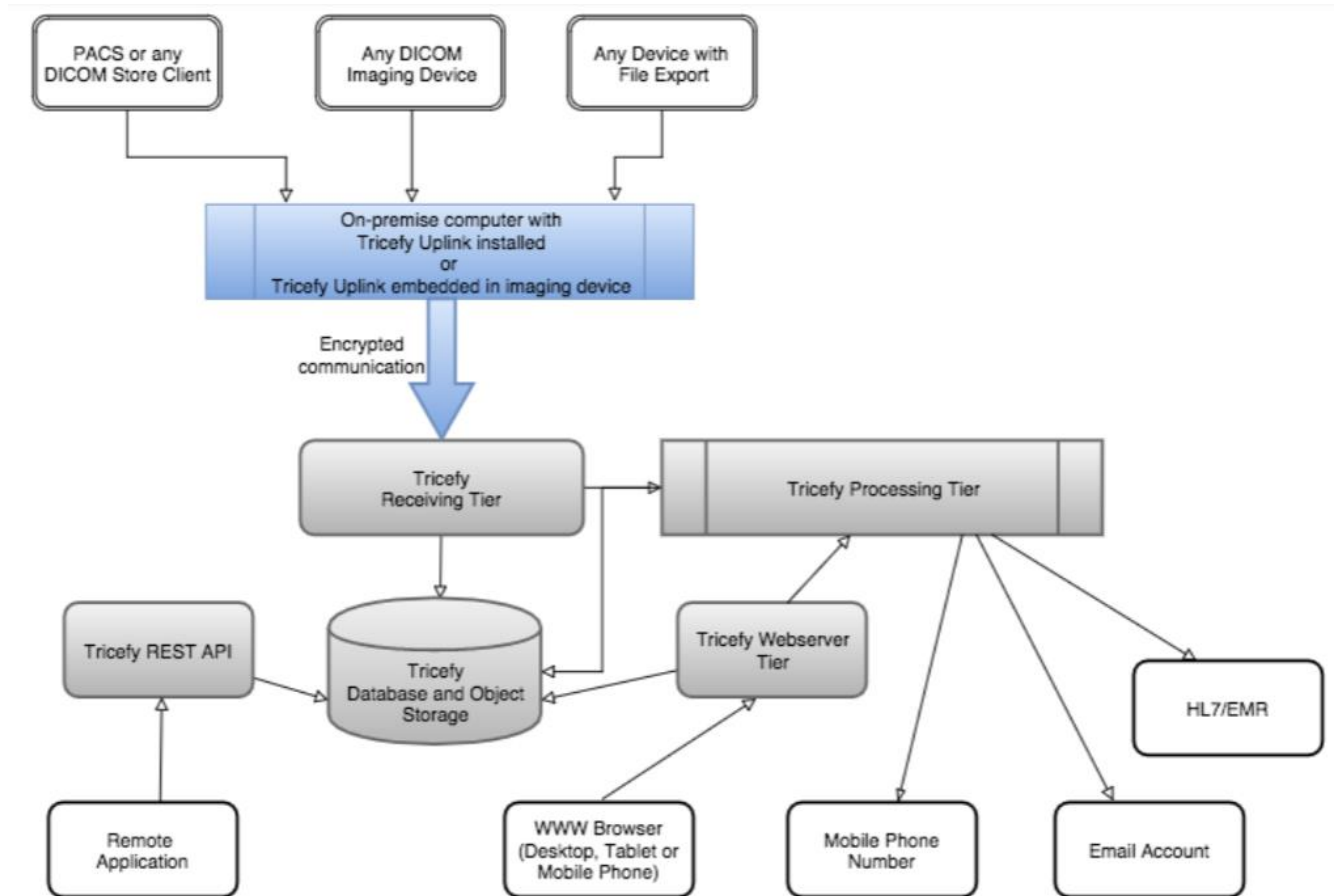


# Tricefy Architecture

This is the DICOM conformance statement for Tricefy and Tricefy Uplink. Tricefy Uplink is the lightweight software agent that runs on customer networks for the purpose of storing images (C-STORE SCP), verifying DICOM communications (C-ECHO SCP), querying for and then retrieving DICOM instances (C-FIND SCP, C-MOVE SCP, C-STORE SCU), and Modality Worklist DICOM queries (C-FIND SCP).

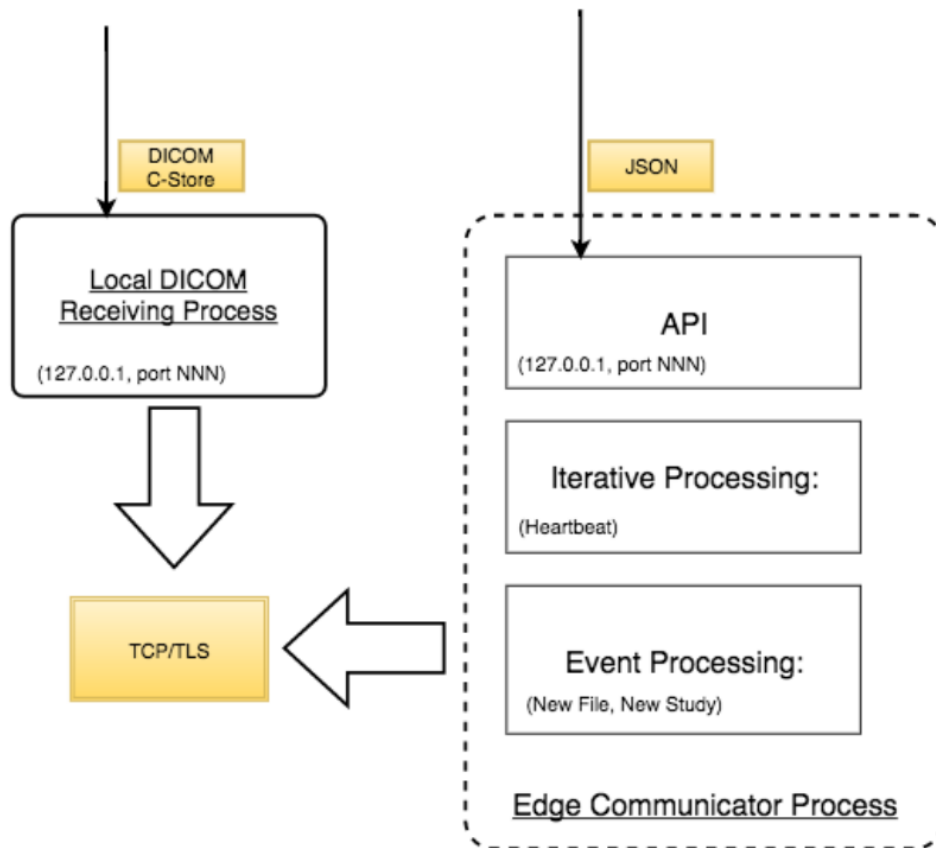
## Tricefy Overview



# Tricefy Uplink Technology

- Runs on any computer in the customer's local network
  - Windows, MAC, Android, Linux
    - Single code base, multiple installers
- Downloaded from URL, easily installed in each environment
- Software automatically updates with new releases
- Dedicated server not needed
- Works well with other apps and services
  - No special monitoring or considerations
- Can be embedded directly in any Ultrasound device
  - Two processes (Android, Linux, MAC) or services (Windows)
  - Small footprint, minimal computing resources:
    - 20M disk for processes and data files
    - Small memory footprint (~ 15M) remaining constant as data is streamed
    - Minimal CPU: older Pentium 4 or dual-core Atom work with moderate traffic

## Tricefy Uplink Processes



## Tricefy Uplink: Local Receiving Process

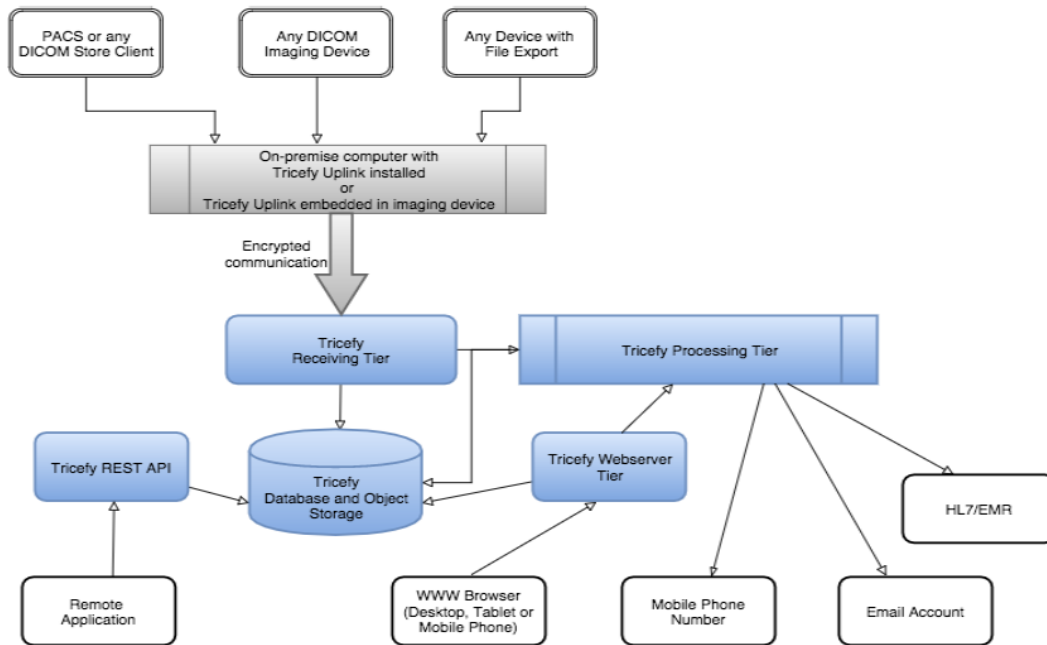
- DICOM Receiver Service
  - A C-STORE, C-ECHO SCP DICOM Receiver (listening on configured port)
    - C-FIND and C-MOVE support (for Modality Worklist and Query/Retrieve)
    - Full DICOM conformance
- C-STORE
  - Send DICOM securely to cloud
    - TLS on TCP
    - Private pre-shared keys (TLS-PSK, device/account specific)
    - AES-256, port 443
    - Robust transmission (will succeed if there is any connectivity at all)
    - Any failure will result in a failed response
    - Success means data received without error and durably queued for processing

## Tricefy Uplink: Edge Communicator Process

- **Edge communication daemon**
  - Iterative processing: processing repeated at time intervals
    - Heartbeat: check in with Receiving tier sending environment details
      - Version, internal IP address, OS version, time zone, etc.
      - Tricefy Uplink Management reports are generated from this information
  - Listen on configured port for API calls expressed as JSON objects
    - These are a sampling of existing request types
    - Very easy to add new request types to the API
      - “request\_type”: “sharing”
        - Initiate Patient or Doctor Sharing
      - “request\_type”: “query authorization”
        - Return if device authorized
      - “request\_type”: “heartbeat”
        - On-demand heartbeat request
      - “request\_type”: “DICOM send”
        - On-demand sending of DICOM from disk file
- **Event processing**
  - DICOM failure event
  - New file (DICOM or non-DICOM) in export directory ready for transfer
  - New study ready for HL7 formatting
  - File ready for download to complete C-MOVE/C-STORE SCU
  - Others to be defined

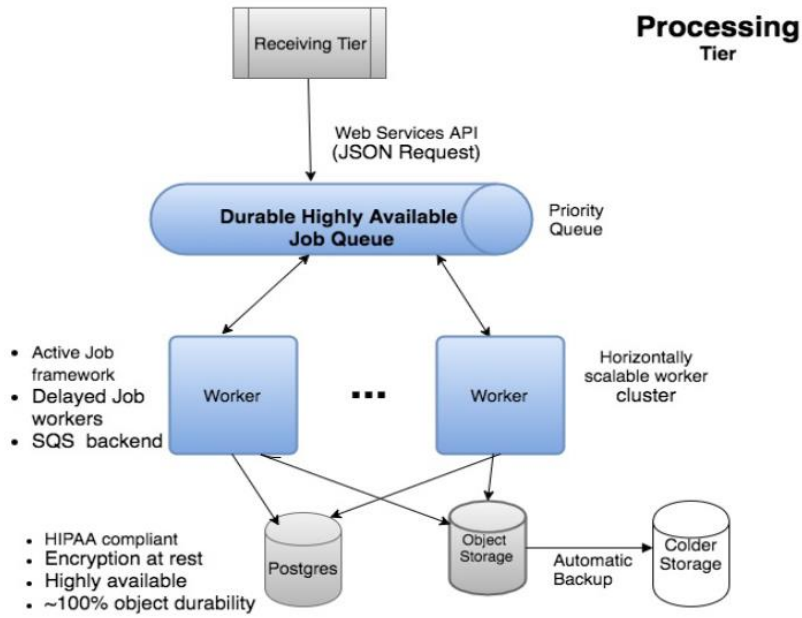
## Tricefy Uplink: Security

- TLS-PSK: <http://tools.ietf.org/html/rfc4279>
  - “pre-shared keys may be more convenient from a key management perspective: it may be easier to configure PSK than to use certificates. A case in point is when the parties already have a mechanism for setting up a shared secret key and that mechanism could be used to ‘bootstrap’ a key for authenticating a TLS connection”
  - DHE\_PSK key exchange algorithm
    - Ciphersuites use a PSK to authenticate a Diffie-Hellman exchange
    - Therefore, additional protections (dictionary attacks, Perfect Forward Secrecy)



## Receiving Tier

- End points for Tricefy Uplink's two customer local services described earlier (Local DICOM Receiver, EdgeCommunicator)
  - TLS, TLS-PSK, AES-256, Port 443
- Receives requests and data from Tricefy Uplink
  - DICOM or other file:
    - Stream directly to object storage (S3)
      - SSL/TLS, private IP addresses
      - AES256 server-side encryption
    - Enqueue processing job to durable redundant queue
      - No PHI enqueued
    - Responds to Tricefy Uplink with proper status to return to imaging device
  - API (JSON) request
  - Heartbeat and other processing
- Multi-threaded, Horizontally scalable service



## Tricefy Database and Object Storage

- Postgres Database
  - HIPAA compliant
    - Security Key Management
    - Encryption at rest including:
      - Backups
      - Read-replicas
      - Snapshots
    - Encrypted database connections
- Object Storage
  - Pre-signed secure URLs to provide temporary user access to secure objects
  - https endpoints for data transport
  - Nearline/Offline storage used for data duplication

## Tricefy Webserver Tier

- Web application
  - Thick client (ember UI)
  - Full API
- Technologies
  - Ruby on Rails
  - Ember.js JavaScript framework
  - Redis caching
- Targets
  - Desktop, tablet, mobile phone

